ON-ROAD VISUAL VEHICLE TRACKING USING MARKOV CHAIN MONTE CARLO PARTICLE FILTERING WITH METROPOLIS SAMPLING

J. ARRÓSPIDE and L. SALGADO

ABSTRACT–In this study, a method for vehicle tracking through video analysis based on Markov chain Monte Carlo (MCMC) particle filtering with metropolis sampling is proposed. The method handles multiple targets with low computational requirements and is, therefore, ideally suited for advanced-driver assistance systems that involve real-time operation. The method exploits the removed perspective domain given by inverse perspective mapping (IPM) to define a fast and efficient likelihood model. Additionally, the method encompasses an interaction model using Markov Random Fields (MRF) that allows treatment of dependencies between the motions of targets. The proposed method is tested in highway sequences and compared to state-of-the-art methods for vehicle tracking, i.e., independent target tracking with Kalman filtering (KF) and joint tracking with particle filtering. The results showed fewer tracking failures using the proposed method.

KEY WORDS : Intelligent vehicles, Image analysis, Object tracking, Monte Carlo methods

1. INTRODUCTION

In the field of intelligent transportation systems (ITS), one of the main challenges is to exploit vehicle sensing capabilities to maximize driver safety. Many studies have been devoted to the use of different sensors to achieve this goal, including radar, laser, LIDAR, video cameras, GPS, and vehicle inertial measurement units (IMUs), e.g., accelerometers, speedometers, odometers and gyroscopes. In recent years, however, video-based systems for scene analysis have received increasing interest from car manufacturers and research centers due to low cost and good performance. Research on object detection and tracking plays an important role because of its ability to address critical issues, such as collision avoidance (Choi *et al.*, 2012; Lee and Kim, 2012).

Video-based object tracking is addressed in the literature using a wide range of approaches. Traditional approaches typically rely on template-based (Asadi *et al.*, 2008; Avidan, 2007; Goecke *et al.*, 2007) or feature-based (Hwang and Huh, 2009; Tran and Davis, 2007; Arróspide *et al.*, 2008) appearance models or motion cues (Sundaresan and Chellappa, 2009).

However, statistical methods, the Bayesian approach in particular, have recently gained increased attention among researchers. The Bayesian approach is exploited in the literature both for independent object tracking and for joint multi-object tracking. The former usually involves Kalman filtering (KF) (Zhao and Nevatia, 2004; Nieto *et al.*, 2011), or any of its variants, such as the extended Kalman filter (EKF) (Barth and Franke, 2009) and the unscented Kalman filter (UKF) (Chen *et al.*, 2009), addressing non-linearities in the objects' motion. In turn, Bayesian tracking of joint multi-target states is commonly formulated with particle filtering (Wang *et al.*, 2008; Khan *et al.*, 2005).

Among probabilistic tracking methods, some authors use a tracking-by-detection approach (Leibe *et al.*, 2008; Wu and Nevatia, 2007; del Blanco *et al.*, 2011), using intra-frame detection applied at every time step. Correspondences in time are then sought by means of association methods, such as JPDA or MHT. Other authors perform the detection once for each vehicle, and then rely solely on tracking. In the field of vehicle imaging, intra-frame detection of vehicles is very time-consuming; therefore, the single-detection approach is usually preferred.

Despite the success of the Bayesian approach for object tracking, especially for surveillance applications, existing methods are limited in two aspects when applied to vehicle tracking in traffic environments. First, interaction between vehicles is usually neglected, which is a major assumption, considering that the trajectory of a vehicle depends to a large degree on the movements of the neighbor vehicles. Second, computational requirements readily become too high to ensure real-time operation. Among existing Bayesian techniques, those based on individual tracking of each object do not include any interaction models and experience frequent tracker failures. In contrast, methods using a joint filter can model interactions, but model complexity grows exponentially with the number of targets (Khan *et al.*, 2005), thus limiting real-time operation.

In this study, a method for multiple vehicle tracking, based on the Bayesian approach, using MCMC sampling to enable joint probabilistic modeling of the vehicle state while achieving real-time operation, is developed. The method also exploits the removed perspective domain given by inverse perspective mapping (IPM) to design effective likelihood and motion models that further reduce the processing cost. The motion model includes an interaction treatment that addresses situations in which the vehicle trajectory is affected by surrounding vehicles.

The proposed method is compared to the two traditional approaches to Bayesian tracking, i.e., independent Kalman filtering (KF) for each vehicle (its variants EKF and UKF do not provide any additional gain as the rectified domain given by the IPM is inherently linear) and joint tracking of the vehicles with a sampling importance resampling (SIR) particle filter. Experiments show that the proposed method results in fewer tracking failures than the traditional methods.

2. GENERAL FRAMEWORK

2.1. Bayesian Approach to State Estimation

The Bayesian approach provides an ideal framework for dynamic state estimation because it recursively updates the state of the system with the new measurements. The Bayesian formulation provides the following expression for the posterior distribution (Arulampalam *et al.*, 2002):

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) = \frac{p(\boldsymbol{z}_k|\boldsymbol{x}_k)\int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1})\mathrm{d}\boldsymbol{x}_{k-1}}{p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1})} \tag{1}$$

where the state of the system at time k is denoted by $\boldsymbol{x}_k$ and the measurements prior to k are given by $\boldsymbol{z}_{1:k} = \{\boldsymbol{z}_i,\ i = 1, \ldots k\}$. The optimal solution of the Bayesian formulation can only be derived analytically when the motion and observation models are linear and the density functions are Gaussian (at which time it is given by the KF). Among suboptimal solutions, particle filters have been widely used due to their simplicity, generality and success over a wide range of challenging applications (Vermaak *et al.*, 2003). Particle filters approximate the posterior density function in Equation (1) with a set of discrete representations, or particles, as:

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) \approx c_1 p(\boldsymbol{z}_k|\boldsymbol{x}_k)\sum_s \pi_{k-1}^{(s)} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(s)}) \tag{2}$$

where $\pi_k^{(s)}$ is the weight associated to a particle $\boldsymbol{x}_k^{(s)}$, and c_1 is the inverse of the evidence factor in the denominator of Equation (1).

2.2. MCMC-based Vehicle Tracking

Figure 1. Steps to segment the image: (a) Original image; (b) Image in the rectified domain; (c) Binary segmented image. Pixels in white correspond to regions segmented as vehicles.

In this study, the use of an MCMC sampling step in the framework of a particle filter, instead of the traditional importance sampling for vehicle tracking, is proposed. The complexity of MCMC increases linearly with the number of objects, as opposed to the complexity of importance sampling (and hence the computational cost), which increases exponentially with the number of objects. This makes MCMC more suitable for applications in which several objects must be tracked simultaneously, as is the case in traffic environments. The MCMC framework also readily models interactions between objects by introducing an additional factor in the probability of the particles. The MCMC-based tracker behaves like a set of individual trackers in the absence of interactions but models relations between objects when they approach each other.

3. PROPOSED METHOD FOR VEHICLE TRACKING

As opposed to tracking-by-detection approaches, the proposed method obtains an initial detection for each vehicle and then relies on vehicle tracking using the MCMC framework described above. To obtain the initial vehicle detection values, the method described by Nieto *et al.* (2011) is used. The method used a two-fold scheme composed of hypothesis generation and hypothesis verification, in which hypotheses are generated by means of a Bayesian classifier using intensity and edge features, and verified according to their symmetry and edge density. The focus of this study is on vehicle tracking; therefore, the reader is directed to Nieto *et al.* (2011) for details on the detection scheme. Any other technique for vehicle detection in the literature can alternatively be used for initialization, such as that proposed by Hoffmann (2006) or Sun *et al.* (2006).

For the tracking algorithm, the proposed method exploits the MCMC framework as follows. The state vector comprising the positions of all the M vehicles in the image, $\boldsymbol{x}_k = \{x_{ik}\}_{i=1}^{M}$, is defined. Note that, at a constant velocity, the evolution of the vehicle positions in the original image space is non-linear, due to the perspective effect introduced by the location of the camera (see example in Figure 1). To avoid the non-linearity, the method uses IPM plane rectification (Bertozzi and Broggi, 1998) to remove perspective and provide a bird's-eye view of the scene (Figure 1(b)). As will be shown in subsequent sections, the rectified domain simplifies the motion and measurement models for vehicle tracking. In the image, vehicles appear distorted in the upper regions because the upper regions are not in the road plane. The analysis will therefore be focused on the lower part of the vehicles because that region conveys the most relevant information due to the high contrast between the road, vehicle shadow and wheels. The position of a vehicle is defined by the coordinates of the mid-lower point in the rectified domain, $\boldsymbol{x}_{ik} = \{\boldsymbol{x}_{ik}, \boldsymbol{y}_{ik}\}$.

Starting with the initial detection, the model objective is to estimate, at each time step, the joint state of the vehicles. The MCMC framework is used to obtain a set of particles (or samples) from the approximation to true data distribution in Equation (2). As opposed to importance sampling, the MCMC uses unweighted samples, reducing Equation (2) to:

$$p(\boldsymbol{x}_k|z_{1:k}) \approx c_2 p(z_k|\boldsymbol{x}_k)\Sigma_s p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(s)}) \tag{3}$$

where $\{\boldsymbol{x}_{k-1}^{(s)}\}_{s-1}^{N}$ are the samples from the previous time step. At each time k, a Markov chain of samples is generated by using the analytic approximation to $p(\boldsymbol{x}_k|z_{1:k})$. The mean of the samples in the previous time step is selected as the initial sample of the chain at time k. Then, to create the chain, candidates, $\boldsymbol{x}_k^{'}$, are subsequently generated by sampling from a proposed distribution, centered on the previous point of the chain, $\boldsymbol{x}_k^{(\tau-1)}$. For each new candidate state of the Markov chain, a single target, j, is selected, for which a new position is hypothesized, using a Gaussian proposed density function, as follows:

$$q(\boldsymbol{x}_{ik}^{'}|\boldsymbol{x}_{ik}^{(\tau-1)}) = N(\boldsymbol{x}_{ik}^{'}|\boldsymbol{x}_{ik}^{(\tau-1)}, \sigma_q)$$

Candidates are accepted or rejected according to the criterion established by the Metropolis algorithm, which defines the acceptance probability as (Bishop, 2006):

$$A(\boldsymbol{x}_k^{'}, \boldsymbol{x}_k^{(\tau-1)}) = \min\left(1, \frac{p(\boldsymbol{x}_k^{'}|z_{1:k})}{p(\boldsymbol{x}_k^{(\tau-1)}|z_{1:k})}\right)$$

The process is simulated by generating a uniform distribution in the interval (0,1) and selecting a number, r, randomly. If $A(\boldsymbol{x}_k^{'}, \boldsymbol{x}_k^{\tau-1}) > r$, then the candidate is accepted, and the following relationships apply:

$$\boldsymbol{x}_{jk}^{(\tau)} = \boldsymbol{x}_{jk}^{'}$$

and

$$\boldsymbol{x}_{ik}^{(\tau)} = \boldsymbol{x}_k^{(\tau-1)} \forall i \neq j.$$

Otherwise, the previous sample is just copied, $\boldsymbol{x}_k^{(\tau)} = \boldsymbol{x}_k^{(\tau-1)}$.

To avoid an unacceptably high correlation between samples, only every L^{th} sample is retained. In this study, L is set at 10. In addition, the first B samples of the chain are discarded (burn-in) to allow the Metropolis algorithm to better approach a stationary distribution, in case of an inaccurate initialization. For this study, B is set at 25.

In the following sections, the likelihood model, $p(z_k|\boldsymbol{x}_k)$, and the system or motion model, $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$, are described. They are used to evaluate the posterior density in Equation (3).

4. LIKELIHOOD MODEL

The segmentation result from Nieto *et al.* (2011) is used as the input to the likelihood model rather than the usual appearance-based approach. The segmentation result uses intensity and gradient features to create vehicle and background classification regions in the rectified domain of the image, given by the IPM (see Figure 1(c)). A new binary image, I_b, is created, in which pixels classified as vehicles are assigned a 1 (white), and those classified as background are assigned a 0 (black). The selected segmentation is designed to be fast and efficient, which is conducive to real-time operation.

Standard appearance-based models applied to the original image are too time-consuming. The relatively straightforward method based on color histograms proposed in Comaniciu *et al.* (2003) has been implemented for comparison, and shown to require more than six times as much processing time as the proposed model (34.18 ms vs. 5.02 ms for tracking). Additionally, as shown in Section 6, color-histogram-based models deliver significantly worse results than the proposed method. More complex appearance-based likelihood models are unaffordable for real-time operation.

The aforementioned model, based on the segmentation result in Nieto *et al.* (2011), is therefore selected for the likelihood model. For a given vehicle, the likelihood of a state is maximum when the state vector coincides with the position observed for that vehicle, i.e., when the hypothesized position lies in the mid-lower point of the white region that represents the vehicle in I_b. Intuitively, if we define a neighborhood centered at $\boldsymbol{x}_{ik}$ in I_b, the likelihood will increase proportionally to the number of white pixels in the upper half of, and decrease proportionally to the number of white pixels in its lower half. More formally, we define the likelihood model of a specific vehicle, $\boldsymbol{x}_{ik} = \{\boldsymbol{x}_{ik}, \boldsymbol{y}_{ik}\}$, as:

$$p(z_k|\boldsymbol{x}_{ik}) = \\ = \frac{1}{F}\left(\sum_{(x,y)\in R_a} I_b(x,y) - \sum_{(x,y)\in R_b} I_b(x,y) + C\right)$$

where R_a is the region above $\boldsymbol{x}_{ik}$, $R_a = \{x_{ik}-\frac{w}{2}\le x\le x_{ik}+\frac{w}{2}\ ;\ y_{ik}-\frac{h}{2}\le y<y_{ik}\}$, where R_b is the region below $\boldsymbol{x}_{ik}$, $R_b = \{x_{ik}-\frac{w}{2}\le x\le x_{ik}+\frac{w}{2}\ ;\ y_{ik}<y<y_{ik}+\frac{h}{2}\}$, and $C=\frac{(w+1)h}{2}$ so that $p(z_k|\boldsymbol{x}_{ik})\ge 0$, and F is the normalization factor, ensuring that $p(z_k|\boldsymbol{x}_{ik})$ integrates to 1.

The joint likelihood model of all vehicles is defined as the product of the individual likelihood terms:

$$p(z_k|\boldsymbol{x}_k) = \prod_i p(z_k|\boldsymbol{x}_{ik})$$

5. MOTION AND INTERACTION MODELS

The motion of vehicles in highways tends to be smooth and can thus be approximated by a linear model with constant local velocity. In the rectified image the distances and velocities are proportional to their real magnitudes, hence the evolution of a vehicle state can be modeled by a linear equation:

$$\boldsymbol{x}_{ik} = \boldsymbol{x}_{i(k-1)} + \Delta\boldsymbol{x}_{ik},$$

where the displacement, $\Delta\boldsymbol{x}_{ik}$, is derived from its previous positions. Some noise must also be considered, to account for small accelerations, hence the motion model for a single vehicle is defined by a Gaussian density function:

$$p(\boldsymbol{x}_{ik}|\boldsymbol{x}_{i(k-1)}) = N(\boldsymbol{x}_{ik}|\boldsymbol{x}_{i(k-1)} + \Delta x_{ik}, \sigma_m).$$

This model is appropriate whenever the vehicles are moving without disturbance; however, it should also be noted that the motion of vehicles is affected when other vehicles appear in their surroundings.

Markov random fields (MRF) constitute a simple but powerful means to model interactions between near objects. If the interaction of vehicles happens in a pairwise fashion, then the joint motion model can be expressed as

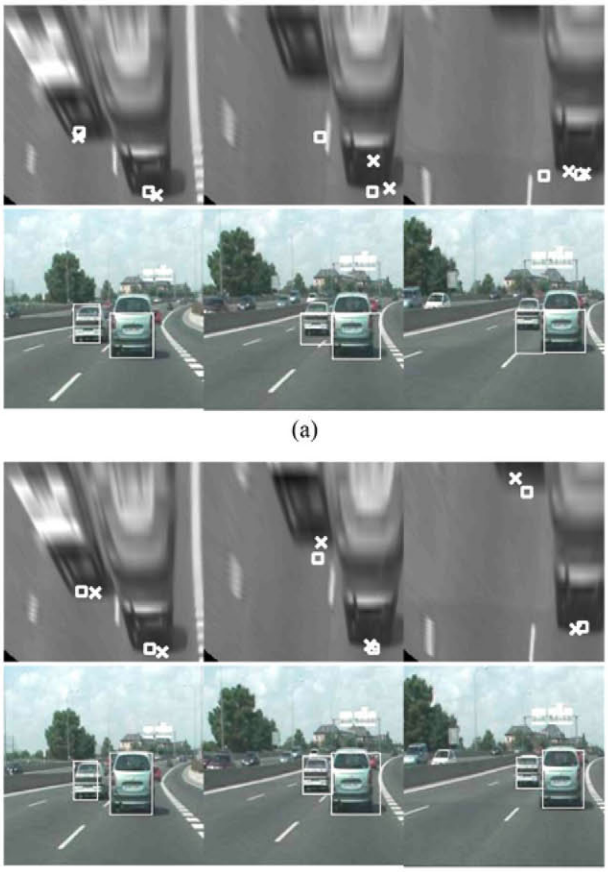
$$p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \prod_i p(\boldsymbol{x}_{ik}|\boldsymbol{x}_{(k-1)})\prod_{ij}\varphi(\boldsymbol{x}_{ik},\boldsymbol{x}_{jk}),$$

where $\varphi(\cdot)$ represents the pairwise interaction potential. The interaction factor is a function of the distance between objects because drivers naturally tend to avoid other cars and to occupy free space. The interaction potential is thus defined as

$$\varphi(\boldsymbol{x}_{ik},\boldsymbol{x}_{jk}) = 1-\varphi(x_{ik},x_{jk})\varphi(y_{ik},y_{jk}),$$

$$\varphi(x_{ik},x_{jk}) = \begin{cases}\exp\left(-\frac{\alpha_x\Delta x^2}{d_x^2}\right) & if\ \Delta x<d_x \\ 1 & \text{otherwise}\end{cases}, \text{ and}$$

$$\varphi(y_{ik},y_{jk}) = \begin{cases}\exp\left(-\frac{\alpha_y\Delta y^2}{d_y^2}\right) & if\ \Delta y<d_y \\ 1 & \text{otherwise}\end{cases},$$



(a)

(b)

Figure 2. Tracking results for an example sequence (a) without interaction model and (b) with interaction model, shown both in the original and the rectified domains. The sequence comprises images at times k_0, k_0+25 and k_0+50.

where $\Delta x = x_{ik}-x_{jk}$ and $\Delta y = y_{ik}-y_{jk}$. The model takes into account the expected security distance both in the driving direction, d_y, and at the sides of the vehicle, $d_x = w_l/4$, where w_l is the width of the lane. The MRF factors operate in such a way that whenever the vehicles in the hypothesized state violate the security distance there is a penalization factor and otherwise $\varphi(\boldsymbol{x}_{ik}-\boldsymbol{x}_{jk})=1$, implying that vehicle j is too far away to affect the driving of vehicle i (and vice versa). The design parameters α_x, and α_y, are tuned so that whenever a vehicle is at half the security distance from another vehicle ($\Delta x = d_x/2$ or $\Delta y = d_y/2$) there is a 0.5 penalization factor.

The interaction factor does not depend on the state in the previous instant; therefore, the approximation to the posterior density in Equation (3) can be rewritten as

$$p(\boldsymbol{x}_k|z_k)\approx$$

$$\approx c_2 p(z_k|\boldsymbol{x}_k)\prod_{ij}\varphi(\boldsymbol{x}_{ik},\boldsymbol{x}_{jk})\sum_s\prod_i p(\boldsymbol{x}_{ik}|\boldsymbol{x}_{i(k-1)}^{(s)})$$

The inclusion of the designed interaction model thus

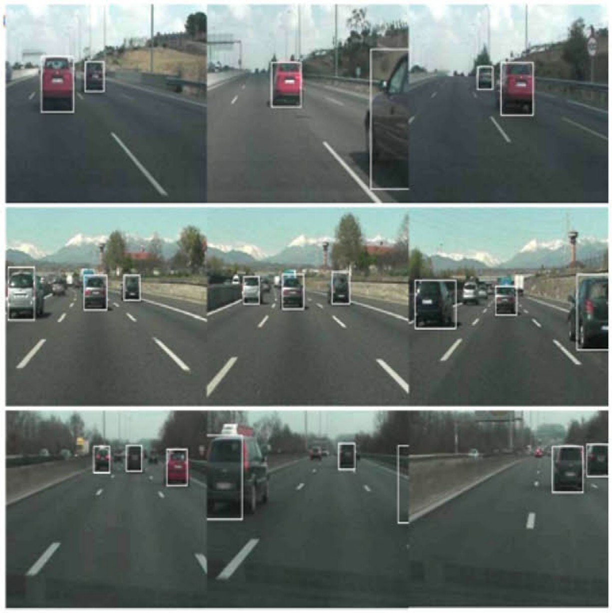


Figure 3. Vehicle tracking for three different sequences (one per row). From left to right, the snapshots show results at times k_0, k_0+130, k_0+230; k_0, k_0+190, k_0+440; k_0+50, k_0+220; and k_0, k_0+80, k_0+240 for sequences 1 to 3 (top-down), respectively.

involves only a marginal increase in the complexity of the posterior density evaluation, while resulting in a significant enhancement of the results, as shown by the tests (see Section 6). The benefits of including this model are illustrated in Figure 2. Tracking results for the method without the interaction model are shown in Figure 2(a), while the results for the proposed method, including the interaction model, are shown in Figure 2(b). Tracking is shown both in the original (lower row) and the rectified (upper row) domains. In the latter, the instantaneous and time-filtered measurements are marked with crosses and square markers, respectively. In the original domain tracking is depicted with boxes bounding the vehicles. In Figure 2(a), the state estimate of the left vehicle is misled by the proximity of another vehicle. Then, both tracks converge to the same vehicle, and the tracking for the left vehicle is lost. In contrast, in Figure 2(b), the high probability of incorrect tracking states is compensated through a penalization factor, preventing the tracking error.

6. RESULTS

The proposed method has been tested for real traffic video sequences featuring a variety of driving situations, including different weather conditions, changing illumination, and varying traffic density. Test sequences were acquired using a forward-looking camera mounted near the rear-view mirror of a vehicle driven on highways, and comprised 21470 frames, containing a total of 77 vehicles. The method operates at 10 fps in C++ on an Intel(R) Core(TM) i5 processor running at 2,67 GHz.

Table 1. Summary of results.

Method	Tracking failures	Number of frames	Number of vehicles
Independent-KF	26	21470	77
Joint-SIR	22		
MCMC-A	35		
Proposed Method	7		

The performance of the proposed method is compared to the two methods typically used in the literature, i.e., independent tracking of objects with Kalman filters (shortly independent-KF), and joint tracking using importance sampling-based PF (joint-SIR). For comparison, the interaction function proposed in Equation (2) is used for joint-SIR (independent-KF cannot accommodate an interaction model) and the same number of samples is used for both, $N = 250$. The design parameters, d_x and d_y, have been set according to the particular setup of the camera, $d_x = 22$, and $d_y = 96$. Other design parameters are $w = 10$, $h = 10$, $\sigma_m = (10,15)$, and $\sigma_q = (5,8)$ for joint-SIR, and $\sigma_q = (2,3)$ for the proposed method. The implementation proposed in Nieto *et al.* (2011) is used to test independent-KF.

In addition to evaluate the strength of the proposed likelihood model, tests are performed using an MCMC framework with a classical likelihood model based on appearance (color histograms have been used as explained in Section 4), denoted MCMC-A in Table 1.

The methods are compared by observing the number of tracking failures when each is applied to the test case. Failures are defined as any situation in which the tracker fails to provide continuous and coherent measures for a given vehicle inside the region of interest (ROI). The ROI is defined as the scope of the IPM, usually consisting of the vehicle's lane and the two adjacent lanes, and extending longitudinally up to a distance, d_f, which depends on the camera calibration. Results are shown in Table 1. As expected, the proposed method largely outperforms the others in terms of tracking failures in the test sequences. Independent-KF cannot accurately track when several targets interact, and the performance of the joint-SIR model worsens as the number of objects increases. The MCMC-A model results in the largest number of tracking failures, showing that the use of simple affordable appearance-based models for the likelihood functions is insufficient. In contrast, the same MCMC scheme using the proposed likelihood model achieves much better performance, which proves the effectiveness of the proposed likelihood model.

Figure 3 shows the results of the proposed tracker for three different sequences. In the first sequence (upper row), the method simultaneously tracks a vehicle that is being rapidly overtaken by a vehicle in the same lane, and another that is changing lanes. In the second, tracking of three vehicles is performed, including a vehicle moving at high speed in the left lane and a vehicle being overtaken in the same lane, before a new dark vehicle is detected to the left. Finally, in the third example, tracking of a new vehicle entering the scene on the near left hand side and moving to the right is shown. At the same time, tracking is maintained for a slow vehicle in the right lane and for a vehicle changing lanes in the far distance.

7. CONCLUSION

In this work, a video-based method for vehicle tracking, based on particle filtering, is proposed that encompasses both efficient sampling and interaction modeling. The method has been shown to perform better than existing methods using the same computational resources. Results reveal that the proposed strategy is able to achieve excellent tracking performance while preserving real-time operation.

ACKNOWLEDGEMENT–This work was supported in part by the Ministerio de Ciencia e Innovación of the Spanish Government under projects TEC2007-67764 (SmartVision) and TEC2010-20412 (Enhanced 3DTV), and co-financed by the Fondo Europeo de Desarrollo Regional FEDER under project PSE-370000-2009-009 (TECMUSA).

REFERENCES

Arróspide, J., Salgado, L., Nieto, M. and Jaureguízar, F. (2008). On-board robust vehicle detection and tracking using adaptive quality evaluation. *Proc. IEEE Int. Conf. Image Processing*, San Diego, CA, USA, 2008-2011.

Arulampalam, S., Maskell, S., Gordon, N. and Clapp, T. (2002). A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Trans. Signal Processing* **50**, **2**, 174–188.

Asadi, M., Monti, F. and Regazzoni, C. S. (2008). Feature classification for robust shape-based collaborative tracking and model updating. *EURASIP J. Image and Video Processing*, Article ID, 274349, 21.

Avidan, S. (2007). Ensemble tracking. *IEEE Trans. Pattern Analysis and Machine Intelligence* **29**, **2**, 261–271.

Barth, A. and Franke, W. (2009). Estimating the driving state of oncoming vehicles from a moving platform using stereo vision. *IEEE Trans. Intelligent Transportations Systems* **10**, **4**, 560–571.

Bertozzi, M. and Broggi, A. (1998). Gold: A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Trans. Image Processing* **7**, **1**, 62–81.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer. New York.

Chen, X., Schonfeld, D. and Khokhar, A. A. (2009). Localization and trajectory estimation of mobile objects using minimum samples. *IEEE Trans. Vehicular Technology* **58**, **8**, 4439–4446.

Choi, H.-C.., Park, J.-M., Choi, W.-S. and Oh, S.-Y. (2012). Vision-based fusion of robust lane tracking and forward vehicle detection in a real driving environment. *Int. J. Automotive Technology* **13**, **4**, 653–669.

Comaniciu, D., Ramesh, V. and Meer, P. (2003). Kernel-based object tracking. *IEEE Trans. Pattern Analysis and Machine Intelligence* **25**, **5**, 564–577.

del Blanco, C. R., Jaureguizar, F. and García, N. (2011). An advanced bayesian model for the visual tracking of multiple interacting objects. *EURASIP J. Advances in Signal Processing* 2011, **130**, 13.

Goecke, R., Pettersson, N. and Pettersson, L. (2007). Towards detection and tracking of on-road objects. *Proc. IEEE Intelligent Vehicles Symp.*, Istanbul, Turkey, 416–421.

Hoffmann, C. (2006). Fusing multiple 2D visual features for vehicle detection. *Proc. IEEE Intelligent Vehicles Symp.*, Tokyo, Japan, 406–411.

Hwang, J. and Huh, K. (2009). Vehicle detection system design based on stereo vision sensors. *Int. J. Automotive Technology* **10**, **3**, 373–379.

Khan, Z., Balch, T. and Dellaert, F. (2005). MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Trans. Pattern Analysis and Machine Intelligence* **27**, **11**, 1805–1819.

Lee, B. and Kim, G. (2012). Robust detection of precending vehicles in crowded traffic conditions. *Int. J. Automotive Technology* **13**, **4**, 671–678.

Leibe, B., Schindler, K, Cornelis, N. and Van Gool, L. (2008). Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Trans. Pattern Analysis and Machine Intelligence* **30**, **10**, 1683–1698.

Nieto, M., Arróspide, J. and Salgado, L. (2011). Road environment modeling using robust perspective analysis and recursive bayesian segmentation. *Machine Vision and Applications* **22**, **6**, 927–945.

Sun, Z., Bebis, G. and Miller, R. (2006). Monocular precrash vehicle detection: Features and classifiers. *IEEE Trans. Image Processing* **15**, **7**, 2019–2034.

Sundaresan, A. and Chellappa, R. (2009). Multicamera tracking of articulated human motion using shape and motion cues. *IEEE Trans. Image Processing* **18**, **9**, 2114–2126.

Tran, S. and Davis, L. (2007). Robust object tracking with regional affine invariant features. *Proc. IEEE Int. Conf. Computer Vision*, Rio de Janeiro, Brazil, 1–8.

Vermaak, J., Doucet, A. and Pérez, P. (2003). Maintaining multi-modality through mixture tracking. *Proc. IEEE Int. Conf. Computer Vision*, Nice, France, 1110–1116.

Wang, J., Yin, Y. and Man, H. (2008). Multiple human tracking using particle filter with gaussian process dynamical model. *EURASIP J. Image and Video Processing*, Article ID 969456, 10.

Wu, B. and R. Nevatia (2007). Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *Int. J. Computer Vision* **75**, **2**, 247–266.

Zhao, T. and Nevatia, R. (2004). Tracking multiple humans in complex situations. *IEEE Trans. Pattern Analysis and Machine Intelligence* **26**, **9**, 1208–1221.